8

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/889,350	07/17/2001	Martin Russ	36-1466	2630

7590 03/23/2005

Nixon & Vanderhye
1100 North Glebe Road 8th Floor
Arlington, VA 22201-4714

EXAMINER	
BLACKWELL, JAMES H	
ART UNIT	**PAPER NUMBER**
2176	

DATE MAILED: 03/23/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	**Application No.** 09/889,350	**Applicant(s)** RUSS ET AL.	
	Examiner James H Blackwell	**Art Unit** 2176	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on 28 October 2004.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) 1-28 is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) 1-28 is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on 17 July 2001 is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☒ All b)☐ Some * c)☐ None of:

1.☒ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date 10/28/04.

4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____.

5) ☐ Notice of Informal Patent Application (PTO-152)

6) ☐ Other: _____.

DETAILED ACTION

This Office Action is in response to Amendment/Response received 10/28/04.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 1-28 are rejected under 35 U.S.C. 103(a) as being unpatentable over Arora et al. (hereinafter Arora, U.S. Patent No. 5,911,145) in view of Guzak et al. (hereinafter Guzak, U.S. Patent No. 5,838,319) and in further view of Hanson et al. (U.S. Patent No. 5,956,736).

In regard to independent Claim 1 (and similarly independent Claims 18-19, and 22), Arora teaches *a method for creating hypermedia content for a web site* in that a user drags and drops display elements to define the hierarchy of the site and to define the layout of each page in the site. The present invention automatically generates a layout for each page. This layout contains display elements that represent the links between pages of the site. Thus, the user does not have to manually specify links for each page. As the user drags and drops icons to add, move, and delete pages of the site hierarchy, the present invention will automatically add, remove, and delete the appropriate links between the pages of the site. A preferred embodiment also automatically creates a banner across the top of each page that contains a user-specified page name. After the user has defined the hierarchy of the site and the layout of each page in the site, the user "publishes" the site. The publish function (*running a conversion program module to convert the file contents into hypermedia for the web site with hyperlinks therein corresponding to the file structure*) automatically generates HTML for each page of the site in accordance with the display elements of each page. In the described embodiment of the present invention, the publish function generates an HTML table for each page. The number of cells in each table reflects the number and placement of display elements on a corresponding page, yielding a true WYSIWYG page for the site (Col. 2, lines 24-44; compare with Claim 1 (and similarly Claims 18-19, and 22), ***"A method of creating hypermedia content for a web site"***). Arora fails to teach *making use of a computer configuration that has an operating system wherein files are configurable in a hierarchical structure and the operating system is provided with a viewer that provides a visual display of the hierarchical structure and an indication of the file content, the method being characterized by using the viewer of the operating system to establish a hierarchical structure of files corresponding to content for the web site*. However, Guzak teaches a method practiced in a computer system that has an output device and a processor for running a first application program. In accordance with this method, a child window control is provided as a system resource. One or more application programs to display a list of items as a hierarchical tree on the output device may use this child window control. The child window control may support the hierarchical tree of items being expandable to show additional levels of items, and/or collapsible to hide levels of items (Col. 1, lines 27-37). Guzak also teaches a hierarchical display used by the operating system that is available to an application program for displaying files hierarchically. Though Guzak does not specifically talk about the type of files to display or the specific application program, it would have been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Arora and Guzak as both contain mechanisms to display a set of files hierarchically. Adding Guzak provides the benefit of performing this task using routines defined within the operating system and thus creating an interface that is more familiar to the user. Arora does not specifically teach a *template library, as taught in Claim 22.* However, Hanson teaches that the object system is based on a prototype-based model where every object can be a template for creating new objects with different types of properties and handlers. In a prototype-based model, no distinction is made between an object and a template, i.e., all objects can be templates for other objects (Col. 7, lines 1-6). In addition, Hanson teaches that the user panel (501), although illustrated as empty in Figs. 5A and 5B, displays new user-defined HTML objects created by the user. It is possible to drag and drop an HTML object from widgets panel (502) to user panel (501). The user may first choose to modify a predefined HTML object from widgets panel (502), rename it, and store it in user panel (501). The predefined HTML object is modified by the user first dragging the HTML object to an object editor window. This allows a user to reuse a predefined HTML object, such as a header, by modifying a property or handler associated with the predefined HTML object, and creating a new user-defined HTML object that is identical to the predefined HTML object with the exception of the modified property or handler (Col. 9, lines 62-67; Col. 10, lines 1-5). It would have therefore been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Arora, Guzak, and Hanson as all three inventions relate to hierarchical displays of files. Adding Hanson provides the benefit of using templates to aid in producing a hierarchical display.

In regard to dependent Claim 2, Claim 2 reflects the method of creating hypermedia content for a web site as Claimed in Claim 1, and is rejected along the same rationale.

In regard to dependent Claim 3, Arora teaches that *the computer configuration is operable to run a plurality of different application programs with individual file formats* in that pages in the site can include new text and images or preexisting text and images (Col. 5, lines 46-47; Fig. 2). Arora also teaches that *the hierarchical structure for the web site includes files in a plurality of said different formats, the method including using the conversion program module to convert the files of the different formats into a form suitable for use as hypermedia on the web* site in that Fig. 7 shows a format of a page object (700). Each page object of Figs. 5(a) and 5(b) has this format. Fig. 7 shows only the information in the object, not the size or type of the information. Page object (700) includes an object number, an object type, an object name (e.g., "Products"), a collection number (currently unused), a number of a parent node/page, a next sibling number, a first child number, a pointer to a list of draw objects in a current layout of the page, a flag indicating whether the page is a stacked page, a flag indicating whether the page is currently expanded on the display, a flag indicating whether the page is currently collapsed on the display, a flag indicating whether the page should be published when the site is published, a color of the page icon for the page, a status (e.g., "done," "not done"), and user-entered comments (Col. 7, lines 31-45). Each page object described by Arora is a template for future web pages created when the Publish button is clicked. Each of the nodes and sub nodes are similar to file folders and subfolders and each are arranged, and can be manipulated in a hierarchical tree structure.

In regard to dependent Claims 4-6, Arora does not explicitly teach that *the hierarchical structure includes a file folder, and at least one underlying layer containing at least one file subfolder.* However, Arora does teach the notion of a page object. Fig. 7 shows a format of a page object (700). Each page object of Figs. 5(a) and 5(b) has this format. Page object (700) includes an object number, an object type, an object name (e.g., "Products"), a collection number (currently unused), a number of a parent node/page, a next sibling number, a first child number, a pointer to a list of draw objects in a current layout of the page, a flag indicating whether the page is a stacked page, a flag indicating whether the page is currently expanded on the display, a flag indicating whether the page is currently collapsed on the display, a flag indicating whether the page should be published when the site is published, a color of the page icon for the page, a status (e.g., "done," "not done"), and user-entered comments (Col. 7, lines 31-45). As taught in Figs. 5a and b, these objects are arranged in a hierarchical structure. The Page objects arranged in this structure could be considered as a set of folders and subfolders containing instructions on how to construct a web page in that location within the hierarchical structure. Arora also teaches that the hierarchy can, but does not necessarily represent links between the pages of the site (Col. 2, lines 20-22). Arora also teaches that after the user has defined the hierarchy of the site and the layout of each page in the site, the user "publishes" the site. The publish function automatically generates HTML for each page of the site in accordance with the display elements of each page (Col. 2, lines 36-40; compare with Claim 4, ***"... and the conversion program module produces web pages corresponding to the folder and any subfolders with hyperlinks between them corresponding to the folder hierarchy"*** and Claim 5, ***"... the folder contains a template file defining a predetermined configuration for hypermedia at a node in the web site corresponding to the folder***" and Claim 6, ***"... the conversion program module produces a web page corresponding to the template for a node corresponding to the subfolder"***).

In regard to dependent Claims 7-8, Arora does not explicitly teach that *the conversion program module searches the subfolder to determine if it contains a template file defining a predetermined configuration for hypermedia at a node in the web site corresponding to the subfolder*. However, considering the Object-Oriented nature of the teachings of Arora, one could assume that standard concepts of OO technology apply (in this case) the notion of inherency. Each page object child (subfolder) is related to its parent (folder), and therefore the child object could inherit traits from its parent. Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to assume that if a template file is found in a subfolder, that it would have been processed, and if it were not found, then the conversion program would have referred up the hierarchy tree until a template was found. The benefit would have been an efficient means of propagating a common design throughout a web site.

In regard to dependent Claims 9-10, Arora does not specifically teach that *the template file includes a plugin for inserting predetermined hypermedia from different files into the web page produced by the template* or that *the plugin defines a link and the conversion program module produces a hyperlink in the web page produced by means of the template with a configuration defined by the link*. However, it would have been obvious to one of ordinary skill in the art at the time of invention to assume that such plugins would simply be included as additional objects to be added to the previously taught Page object (Fig. 7), rendered into an action once the Page object was instantiated by invoking the Publish button, as taught by Arora. The benefit would have been to save time by incorporating sources of information that had previously been published.

In regard to dependent Claim 11, Claim 11 reflects the method of creating hypermedia content for a web site as Claimed in Claim 4, and is rejected along the same rationale.

In regard to dependent Claim 12, neither Arora nor Guzak teach *providing a library of said templates and using the file viewer to transfer a selected one or more of the templates from the library to the one or more of the folders*. However, Hanson teaches that the object system is based on a prototype-based model where every object can be a template for creating new objects with different types of properties and handlers. In a prototype-based model, no distinction is made between an object and a template, i.e., all objects can be templates for other objects (Col. 7, lines 1-6). In addition, Hanson teaches that the user panel (501), although illustrated as empty in

Figs. 5A and 5B, displays new user-defined HTML objects created by the user. It is possible to drag and drop an HTML object from widgets panel (502) to user panel (501). The user may first choose to modify a predefined HTML object from widgets panel (502), rename it, and store it in user panel (501). The predefined HTML object is modified by the user first dragging the HTML object to an object editor window, as described in more detail below. This allows a user to reuse a predefined HTML object, such as a header, by modifying a property or handler associated with the predefined HTML object, and creating a new user-defined HTML object that is identical to the predefined HTML object with the exception of the modified property or handler (Col. 9, lines 62-67; Col. 10, lines 1-5). It would have therefore been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Arora, Guzak and Hanson as all three inventions relate to the hierarchical display of files. The addition of Hanson provides the benefit of an efficient mechanism for the construction of Web documents

In regard to dependent Claim 13, Arora teaches in Fig. 8 a flow chart showing initialization steps performed by structure editor software (120) upon execution. Step (802) opens the initial window of Fig. 3. Step (804) sets default values for the site in memory. These default values include a style of the layout, header, and footer, as discussed below. Step (806) creates and displays a page object for the homepage (root node) of the site in memory (104). The created page object has a format shown in Fig. 7 and has no parent node. Step (808) stores default initial values in the root page object (Col. 7, lines 46-55). In addition, Arora teaches in Fig. 9 a flow chart showing steps performed by structure editor software (120) to create a new page in the site. The steps of Fig. 9 are performed when the user selects New Page button (314) of Fig. 3. If, in step (902), the user has previously selected a page icon (also called a "node"), then processor (102) displays a new child page icon of the selected page icon in step (904). Step (906) creates a page object in memory for the new page. If the selected node already has an existing child then, in step (910), the new node is made a next sibling of the existing child node. Otherwise, in step (912), if the selected node has no children, then the new node is made a first child of the selected node. In the described embodiment, a new page is always a normal page (Col. 7, lines 66-67; Col. 8, lines 1-11; compare with Claim 13, ***"... the folder structure includes a root folder and sub-folders depending therefrom, and including placing at least one of the templates in the root folder"***).

In regard to dependent Claim 14, Arora teaches in Fig. 1 a computer system (100) that includes, among other things, network connections, LANs, Internet connections, input/output lines, etc. (Col. 5, lines 30-34; compare with Claim 14, ***"... the computer configuration comprises a network"***). Arora does not specifically teach *arranging the files in the hierarchical structure for the web site, using the file viewer, from different file locations in the network*. However, Arora does teach that pages in the site can include new text and images or preexisting text and images (Col. 5, lines 46-47). It would have been obvious to one of ordinary skill in the art at the time of invention to assume that pages in the site could come from elsewhere, providing the benefit of saving time incorporating material into the web site by using material already created elsewhere.

In regard to dependent Claim 15, Arora does not explicitly teach *uploading the hypermedia for installation on a server for the web site*. However, Arora teaches a Publish button that creates HTML files from the Page objects (Fig. 3). In addition, Arora teaches in Fig. 1 a computer system 100 that includes, among other things, network connections, LANs, Internet connections, input/output lines, etc. (Col. 5, lines 30-34). Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to use the network connection to upload HTML files to a server, providing the benefit of making web site material available to the World-Wide-Web.

In regard to dependent Claim 16, Arora teaches a Preview Button (Fig. 3; compare with Claim 16, ***"... providing a local preview of the hypermedia before uploading it to the server"***).

In regard to dependent Claim 17, Arora does not specifically teach *causing the conversion program module to be downloaded to the computer configuration from a remote server*. However, Arora does teach in Fig. 1 a computer system 100 that includes, among other things, network connections, LANs, Internet connections, input/output lines, etc. (Col. 5, lines 30-34). It would have been obvious to one of ordinary skill in the art at the time of invention to make the conversion program available for download, using the network connection, providing the benefit of a software tool for designing and creating a web site.

In regard to dependent Claim 20, Arora teaches in Fig. 1 an input device (161) such as a floppy drive or a CD ROM reader that reads computer instructions and data stored on computer readable medium (162) such as a floppy disk or a CD ROM. These computer instructions are the instructions of e.g., structure editor software (Col. 5, lines 18-23; compare with Claim 20, ***"A program module recorded on a recording medium insertable into the computer configuration to be loaded therein"***).

In regard to dependent Claim 21, Arora does not specifically teach *a program module configured as a download to be downloaded from a server to the computer configuration*. However, Arora does teach in Fig. 1 a computer system 100 that includes, among other things, network connections, LANs, Internet connections, input/output lines, etc. (Col. 5, lines 30-34). It would have been obvious to one of ordinary skill in the art at the time of invention to make the conversion program available for download, using the network connection, providing the benefit of a software tool for designing and creating a web site.

In regard to independent Claims 23 and 24, Claims 23 and 24 reflect the method of creating hypermedia content for a web site as claimed in Claim 1, and are rejected along the same rationale.

In regard to dependent Claims 25-28, Arora fails to explicitly teach that *the operating system is executed in a networked environment for concurrent access by multiple users*. However, Arora does teach the notion of "publishing" a web site wherein HTML pages are generated for a site (presumably a web site). Web sites are inherently known to be accessible to multiple simultaneous users. Hanson also speaks of publishing HTML pages to a web site where the web site is on the Internet. The web site (server) acts with a web browser (client) in which one or more clients 810 and 820 access or query a server 830 over a data communications network (Col. 2, lines 56-67; Col. 3, lines 1-22; Col. 13, lines 51-57). It would have been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Arora and Hanson as both inventions deal with creating web sites. Adding Hanson provides the benefit of publishing to a web server that is capable of simultaneous multiple user accesses.

Response to Arguments

Applicant's arguments, see Remarks/Arguments, filed 10/28/04, with respect to the rejection(s)of claim(s) 1-22 under Arora in view of Hanson have been fully considered and are persuasive. Therefore, the rejection has been withdrawn. However, upon further consideration, a new ground(s) of rejection is made in view of Guzak et al. (hereinafter Guzak, U.S. Patent No. 5,838,319). Specifically, applicant argues that the combination of Arora and Hanson *fails to teach or suggest an operating system provided with a viewer that provides a visual display of a hierarchical structure of files and an indication of file content as required by independent claim 1 and its dependents. In dependent claims 18, 19, and 22 require similar limitations*. The examiner agrees that the combination of Arora and Hanson fails to teach the limitation as claimed in Claim 1 (and similarly in Claims 18, 19, and 22). However, examiner has added the reference of Guzak, which teaches a method practiced in a computer system that has an output device and a processor for running a first application program. In accordance with this method, a child window control is provided as a system resource. One or more application programs to display a list of items as a hierarchical tree on the output device may use this child window control. The child window control may support the hierarchical tree of items being expandable to show additional levels of items, and/or collapsible to hide levels of items (Col. 1, lines 27-37). Guzak also teaches a hierarchical display used by the operating system that is available to an application program for displaying files hierarchically. Though Guzak does not specifically talk about the type of files to display or the specific application program, it would have been obvious to one of ordinary skill in the art at the time of invention to combine the teachings of Arora and Guzak as both contain mechanisms to display a set of files hierarchically. Adding Guzak provides the benefit of performing this task using routines defined within the operating system and thus creating an interface that is more familiar to the user.

Applicant argues that the hierarchical structure editor of Arora *does not utilize a viewer provided with the operating system for providing a visual display of the operating system's hierarchical structure of files and an indication of its file content*. The examiner agrees. However, for reasons argued above, the combination of Arora and Guzak would have been obvious to one of ordinary skill in the art at the time of invention to combine Guzak with Arora because both of these inventions contain mechanisms to display a set of files hierarchically. The addition Guzak's teaching would have made it possible to incorporate the operating system's existing method of displaying a hierarchical file structure into other applications providing the benefit of an application more familiar to a user of the existing operating system.

Applicant also argues that Arora is similar to the acknowledged prior art commented on in the applicant's specification (see p. 2, lines 6 to p. 3, lines 6). The examiner agrees. However, the combination of Arora and Guzak would have been obvious to one of ordinary skill in the art at the time of invention because both of these inventions contain mechanisms to display a set of files hierarchically. The addition Guzak's teaching to that of Arora would have made it possible to incorporate the operating system's existing method of displaying a hierarchical file structure into other applications providing the benefit of an application more familiar to a user of the existing operating system.

Conclusion

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to James H Blackwell whose telephone number is 571-272-4089. The examiner can normally be reached on Mon-Fri.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Joseph H Feild can be reached on 571-272-4090. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

James H. Blackwell
03/18/05

SANJIV SHAH
PRIMARY EXAMINER